

Pemodelan Basis data Graph dengan Neo4j (Studi Kasus : Basis Data Sistem Informasi Penjualan pada UMKM)

Muhammad Sholeh
Program Studi Informatika
Fakultas Teknologi Industri
Institut Sains & Teknologi AKPRIND
Yogyakarta
muhash@akprind.ac.id

RR. Yuliana Rachmawati
Program Studi Informatika
Fakultas Teknologi Industri
Institut Sains & Teknologi AKPRIND
Yogyakarta
yuliana@akprind.ac.id

Erma Susanti
Program Studi Informatika
Fakultas Teknologi Industri
Institut Sains & Teknologi AKPRIND
Yogyakarta
erma@akprind.ac.id

Abstract— This research implements data storage using database graph. Graph database is one of the categories of the noSQL database. In the database the SQL data model is formed in tables consisting of rows and columns, whereas in the database the NoSQL data model does not have a standard scheme that must be defined. NoSql is a database management system that does not have or comply with certain rules as in the relational database management system model. NoSQL has a dynamic schema while in SQL database it follows the schema that was set at the beginning of the design. The development of MSME sales system database graph is done by developing conceptually first and comparing it by making a relational database model and a graph, graph model first. The results of this study produce a graph database that manages sales data and connects various nodes that are formed and the query process to produce information. Implementation of a graph database database model uses the Neo4j database

Keywords— database, graph, noSql, sales, SQL,

Abstrak— Penelitian ini mengimplementasikan penyimpanan data dengan menggunakan basis data graph. Basis data graph merupakan salah satu kategori dari basis data noSQL. Dalam basis data model SQL data dibentuk dalam tabel –tabel yang terdiri dari baris dan kolom, sedangkan pada basis data model NoSQL data tidak memiliki skema standar yang harus didefinisikan. NoSql merupakan sistem manajemen basis data yang tidak mempunyai atau mematuhi aturan tertentu seperti pada model sistem manajemen basis data relasional. NoSQL memiliki skema yang dinamis sedangkan pada basis data SQL mengikuti skema yang telah ditetapkan pada awal perancangan. Pengembangan basis data graph sistem penjualan UMKM ini dilakukan dengan mengembangkan terlebih dahulu dalam konseptual dan membandingkan dengan membuat terlebih dahulu model basis data relasional dan model basis data graph. Hasil penelitian ini menghasilkan basis data graph yang mengelola data-data penjualan serta menghubungkan berbagai simpul yang terbentuk serta proses query untuk menghasilkan informasi. Implementasi dari model basis data basis graph menggunakan basis data Neo4j

Keywords— basis data, graph, noSql, penjualan, SQL,

PENDAHULUAN

Dalam pengembangan aplikasi sistem informasi, peranan basis data sangat penting. Dalam basis data ini semua data-data yang diolah dalam sistem informasi disimpan. Model basis data yang saat ini populer adalah penggunaan basis data relasional. Dalam pembuatan basis data relasional, data-

data disimpan dalam bentuk tabel-tabel dan tabel-tabel tersebut dihubungkan dalam bentuk relasional. Basis data relasional menggunakan relasi untuk menyimpan data dan mencocokkan data. Proses untuk pengelolaan data-data yang tersimpan dalam basis data menggunakan pemrograman basis data yaitu SQL (Structure Query Language) . Kelemahan penggunaan basis data relasional diantaranya adalah pertumbuhan data yang sangat besar dan kebutuhan struktur data yang berbeda-beda pada setiap data berakibat basis data yang dibangun mengalami kesulitan. SQL adalah bahasa standar yang digunakan untuk memanipulasi dan mengambil data dari suatu basis data relasional. SQL dapat digunakan melakukan pengelolaan data-data seperti memodifikasi struktur basis data, mengubah pengaturan keamanan sistem , menambahkan izin pengguna pada basis data atau tabel , mengatur permintaan data untuk informasi serta melakukan pembaharuan pada konten basis data.[1]

Model basis data relasional digunakan untuk menyimpan, memanipulasi dan mengambil data. Salah satu kelemahan dari model basis data relasional adalah masalah kecepatan dan daya tampung terhadap data yang besar. Kecepatan data dalam proses query akan menurun karena pertumbuhan data. Upaya untuk mengatasi pertumbuhan dan kecepatan query data adalah dengan menggunakan basis data kategori nosql grafik dan salah satu model nosql menggunakan basis data graph basis data dapat digunakan untuk menyimpan data yang saling terhubung. Dalam artikel ini, dilakukan perbandingan antara basis data relasional dan grafik basis data dengan referensi percobaan dilakukan. [2].

Kelemahan dari basis data relasional memunculkan konsep noSQL. Perkembangan noSQL tidak lepas dari kesulitan-kesulitan yang terjadi dalam penanganan perkembangan basis data dengan menggunakan konsep basis data relasional. Basis data tidak hanya berkembang secara vertikal (adanya penambahan baris), tetapi juga berkembang secara horizontal (adanya penambahan field). Ketika field bertambah banyak permasalahan yang terjadi ketika menggunakan konsep Basis data. Dalam konsep basi data relasional , penambahan field merupakan masalah yang harus dihindari, akan tetapi pada kenyataannya hal itu sering terjadi. Konsep tersebut yang mendasari adanya konsep NoSQL. NoSQL tidak membutuhkan skema tabel dan umumnya menghindari operasi join karena berkembang secara horizontal. Berbeda dengan sistem manajemen basis data relasional, sebagian besar basis data NoSQL dirancang

untuk berkembang dengan baik dan tidak bergantung pada perangkat keras yang tersedia.[3]

Penelitian yang dilakukan Bhaswara, [4], melakukan uji coba dengan membandingkan antara penggunaan SQL dan noSQL terutama dalam ERO Retail. Kesimpulan dalam penelitian ini, basis data NoSQL terbukti memiliki kecepatan penyimpanan data yang lebih unggul dalam hal CRUD daripada SQL. Juga memiliki struktur penyimpanan data yang fleksibel karena model data berupa BSON (Binary JSON). Dan memiliki kemampuan untuk menjadi scalable dengan metode *sharding*. Jadi dalam hal ini basis data NoSQL akan lebih baik untuk diterapkan pada ERP Retail. Penelitian yang membandingkan SQL dan noSQL yang lain dilakukan [5],[6],[7]

Cara kerja basis data noSQL dengan menggunakan berbagai model basis data untuk mengelola dan mengakses data, seperti dokumen, key-value, grafik, in-memory dan search-engine. Jenis basis data NoSQL diantaranya mongoDb, CouchDB, Cassandra, Neo4j.

Basis data graph menggunakan model data topografi untuk menyimpan data. Basis data ini menghubungkan titik data spesifik dan membuat hubungan dalam bentuk grafik yang kemudian dapat digunakan untuk mengolah informasi dengan query. Node dapat mewakili pelanggan, perusahaan, atau data apa pun yang dipilih perusahaan untuk disimpan. Menurut Sasaki [8], dengan menggunakan basis data graph persoalan-persoalan yang muncul dalam SQL seperti perkembangan volume data yang semakin meningkat yang dapat menimbulkan persoalan dalam sisi kecepatan untuk mendapatkan informasi dengan basis data serta fleksibilitas dalam perubahan struktur.

Basis data jenis grafik menggunakan node sebagai entitas data dan edge sebagai hubungan antar entitas. Setiap edge memiliki node awalan dan node akhiran. Edge juga menggambarkan hubungan antara orang tua-anak, kepemilikan, tindakan dan lain sebagainya. Tidak ada batasan jumlah suatu node untuk terhubung dengan node lainnya.[9]. Penelitian mengenai analisis penggunaan graph dilakukan [10], [11],[12],[13]

Penelitian yang berhubungan dengan basis data graph terutama Neo4j diantaranya dilakukan Widyayanti [14], basis data Graph digunakan untuk penyimpan basis data hotel terutama pengelolaan kamar. Zhang [15], membuat studi kasus ini dengan menggunakan basis data grafik Neo4j; salah satu model data NoSQL, untuk membangun mini prototipe blog untuk melakukan rekomendasi sosial yang efisien. Ataky [16], menyimulasikan aplikasi perencana rute kereta api serta menyimulasikan jaringan rel stasiun kereta api yang terhubung satu sama lain di mana setiap koneksi antara dua stasiun mungkin memiliki beberapa properti, query yang dilakukan dapat menemukan bagaimana rute (jalur) yang dioptimalkan dan mengetahui apakah suatu stasiun dapat dijangkau dari satu stasiun atau tidak. Penelitian lain yang membangun basis data dengan Neo4j diantaranya [17], [18],[19].

Hasil penelusuran pustaka yang dilakukan, penelitian yang dilakukan penelitian sebelumnya diantaranya penggunaan basis data non relasional disamping menggunakan model graph, peneliti lain menggunakan mongoDb, Casandra dan lainnya. Peneliti yang menggunakan basis data model Graph dengan Neoj

mengupas selain basis data UMKM seperti basis data hotel, kendaraan.

Pengembangan sistem informasi penjualan untuk pelaku UMKM sudah dilakukan penelitian sebelumnya [20] dan [21]. Masalah dari penelitian sebelumnya diantaranya pengembangan basis data menggunakan model relasional dengan struktur yang sama untuk setiap data pelaku UMKM, sehingga memunculkan masalah dalam proses penyimpanan data di basis data. Hal ini disebabkan, data yang ada dalam UMKM tidak bisa diseragamkan untuk setiap UMKM. Setiap UMKM mempunyai struktur data yang berbeda terutama untuk data pelaku UMKM dan informasi produk yang dijual. Solusi yang dilakukan adalah dengan mengembangkan basis data model noSql yang dapat mengatasi data-data yang tidak struktur dengan model basis data graph dan menggunakan Neo4j.

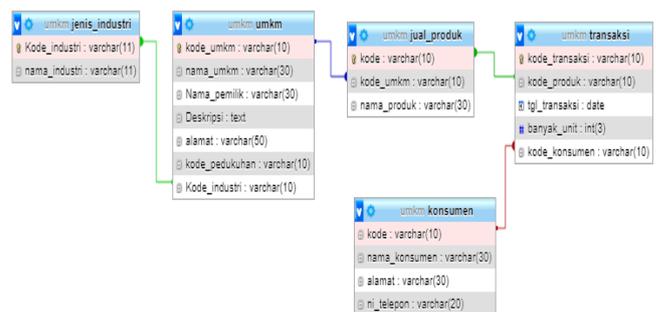
Tujuan dari pengembangan basis data graph untuk mengelola data sistem informasi penjualan UMKM ini diantaranya untuk mendapatkan hasil sejauh mana penggunaan basis data non relasional dengan menggunakan basis data nosql terutama dalam penggunaan basis data UMKM yang terdapat data yang tidak terstruktur.

Rumusan masalah penelitian adalah bagaimana pengembangan basis data penjualan dengan menggunakan basis data Graph, bagaimana membuat simpul-simpul dengan data-data yang ada pada basis data relasional yang sudah dibangun dengan menggunakan basis data Neo4j.

METODE PENELITIAN

A. Basis Data Relasional

Dalam pengembangan basis data graph ini langkah awal mengembangkan basis data relasional. Desain basis relasional ada pada gambar 1 dan selanjutnya dilakukan analisis dan desain bagaimana data yang ada dalam basis data relasional tersebut dikembangkan dalam bentuk basis data graph.



Gambar 1 Basis Data Relasional

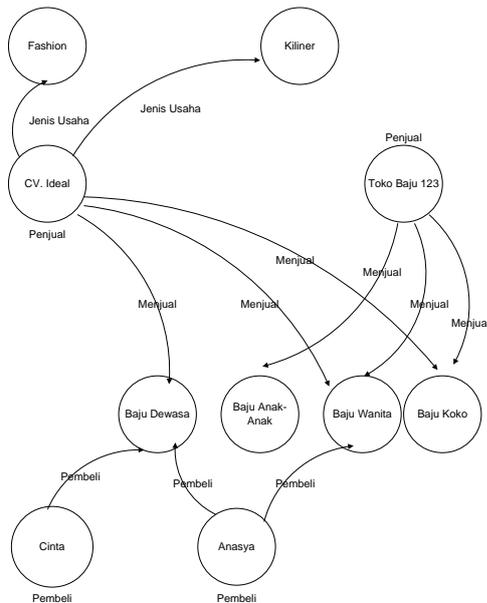
Berdasar relasi pada gambar 1, tabel tersebut dibedakan menjadi tabel master yaitu tabel jenis industri, tabel UMKM dan tabel konsumen dan tabel transaksi yaitu tabel jual_produk dan tabel penjualan. Dari tabel-tabel tersebut dilakukan pembuatan basis data graph dengan membuat simpul-simpul itu yaitu simpul-simpul yang berisi simpul jenis_industri, simpul UMKM dan simpul konsumen. Berbeda dengan basis data relasional, data-data yang ada dalam tabel disimpan dalam bentuk record, dalam basis data graph data-data tersebut disimpan dalam bentuk simpul.

Proses menampilkan informasi yang diinginkan menggunakan bahasa pemrograman Basis data SQL. Perintah SQL membuat tabel transaksi

```
CREATE TABLE `transaksi` (
  `kode_transaksi` varchar(10) NOT NULL,
  `kode_produk` varchar(10) NOT NULL,
  `tgl_transaksi` date NOT NULL,
  `banyak_unit` int(3) NOT NULL,
  `kode_konsumen` varchar(10) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

B. Rancangan Konseptual Basis Data Graph

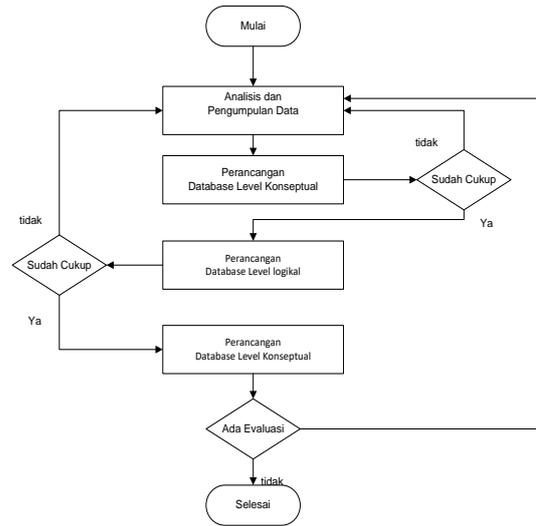
Pada rancangan basis, data graph, masing-masing data dibuat dalam bentuk lingkaran yang mewakili sebuah simpul (*node*). Setiap *node* terhubung ke *node* lain melalui garis yang memiliki arah yang disebut sebagai *edge*. Baik simpul maupun *edge* memiliki *properties*. Berkebalikan dari basis data relasional, nilai *properties* pada *node* nilai *node* atau *edge*. Gambar 2, Rancangan konseptual basis data graph. Node yang ada terdiri dari jenis_industri, penjual(pelaku UMKM), pembeli dan barang yang dijual. Di dalam node terdapat properti yang tiap node bisa berbeda-beda.



Gambar 2 Rancangan Basis Data Graph

B. Tahapan Penelitian

Tujuan perancangan basis data diantaranya untuk memenuhi informasi yang berisikan kebutuhan-kebutuhan user. Basis data merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, digunakan perangkat lunak tertentu untuk memanipulasinya. Basis data merupakan salah satu komponen yang penting dalam system informasi, karena berisi data untuk penyediaan informasi bagi para pemakainya. Gambar 3 tahapan penelitian terutama dalam pembuatan basis data yang dilakukan.

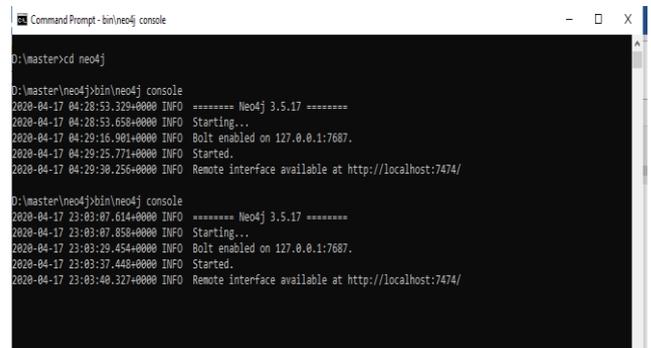


Gambar 3 Tahapan pembuatan Basis data

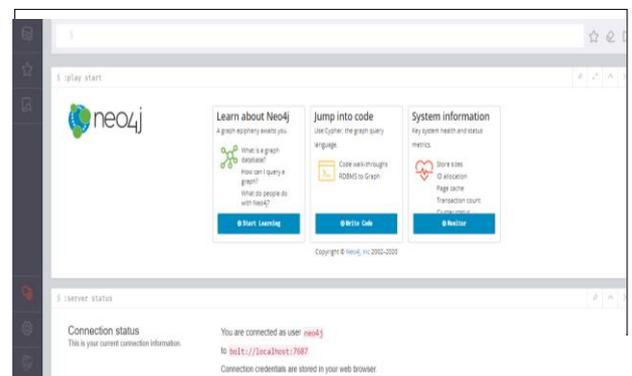
HASIL DAN PEMBAHASAN

A. Menjalankan Neo4j

Pengembangan basis data graph dengan Neo4j diawali dengan mengaktifkan neo4j melalui perintah console. Gambar 4 proses menjalankan neo4j melalui console dan setelah aktif proses pembuatan basis data graph dapat dilakukan melalui browser dengan pemanggilan melalui browser dengan link localhost:7474. Gambar 5. hasil laman browser yang digunakan untuk mengetikkan [perintah-perintah query.



Gambar 4 Proses mengaktifkan Neo4j



B. Membuat simpul dan hubungan

Dalam pembuatan simpul, setiap record yang ada dalam tabel dalam basis data relasional harus dibuat satu-persatu. Dalam basis data graph tidak mengenal record atau tabel, setiap data yang ada dibuat suatu simpul dan hubungan antar simpul dapat dibuat penghubung antar simpul

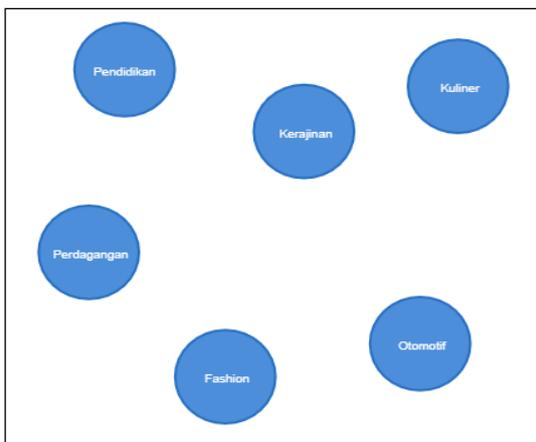
1. Membuat simpul kategori

Simpul ini menyimpan data-data yang terkait dengan kategori nama usaha dari pelaku UMKM. Properti ini minimal terdiri dari Id_kategori dan nama_kategori

- Pembuatan simpul-simpul kategori

```
$create (:kategori
{id_kategori:1,nama_kategori:'Fashion' })
$create (:kategori
{id_kategori:2,nama_kategori:'Kuliner' })
$create (:kategori
{id_kategori:3,nama_kategori:'Otomotif' })
$create (:kategori
{id_kategori:4,nama_kategori:'Pendidikan' })
$create (:kategori{id_kategori:5,nama_kategori:'Perdagangan'})
$create (:kategori {id_kategori:6,nama_kategori:'Kerajinan'})
```

- Hasil proses query tersebut ditampilkan pada gambar 6
\$ MATCH (n:kategori) RETURN n LIMIT 25



Gambar 6 Simpul Kategori

2. Pembuatan Simpul UMKM

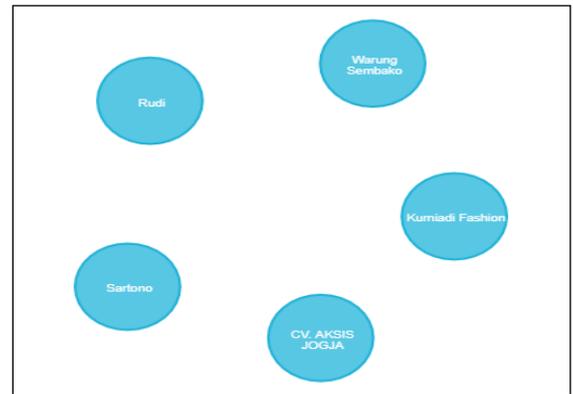
Simpul ini menyimpan data-data yang terkait dengan nama pelaku UMKM. Simpul ini minimal terdiri dari kode_umkn, ID-kategori, pemilik, deskripsi, nama_umkn dan alamat

- Pembuatan simpul-simpul kategori


```
$create (:UMKM{ id_umkm:'u1', pemilik:'Agung',
nama_umk:'CV. AKSIS JOGJA',
deskripsi:'Memproduksi baju' })
$create (:UMKM{ id_umkm:'u2', pemilik:'Kurniadi',
nama_umk:'Kurniadi Fashion', deskripsi:'Menjual baju' })
$create (:UMKM{ id_umkm:'u3', pemilik:'Vina',
nama_umk:'Warung Sembako', deskripsi:'Jual Sembako'})
```

```
$create (:UMKM{ id_umkm:'u4', pemilik:'Rudi',
nama_umk:'Rudi', deskripsi:'Kerajinan Kasongan'})
$create (:UMKM{ id_umkm:'u5', pemilik:'Sartono',
nama_umk:'Sartono', deskripsi:'Kerajinan Bambu'})
```

- Hasil proses query tersebut ditampilkan pada gambar 7



Gambar 7 Simpul UMKM

3. Pembuatan simpul barang

Simpul ini menyimpan semua data-data yang terkait dengan data-data produk yang dijual. Simpul ini minimal terdiri dari id-produk, id_umkm, nama_produk dan harga

- Pembuatan Simpul barang


```
$create (:barang {id_produk : 'p1',
nama_produk:'Baju Anak-anak', harga:50000 })
$create (:barang {id_produk : 'p2',
nama_produk:'Baju dewasa', harga:50000 })
```

- Hasil proses query tersebut ditampilkan pada gambar 8



Gambar 8 Simpul barang

4. Pembuatan simpul pembeli

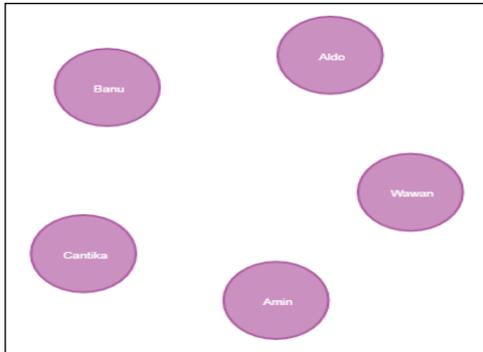
Simpul ini menyimpan semua data-data yang terkait dengan data-data pembeli. Simpul ini minimal terdiri dari kode_konsumen, nama pembeli dan alamat

- Pembuatan Simpul jual_produk


```
$create (:Pembeli {id_member:1, nama:'Amin',
alamat:'Tamantirto'})
$create (:Pembeli {id_member:2, nama:'Wawan',
alamat:'Jl.Bibis 23' })
```

```
$create (:Pembeli {id_member:3, nama:'Aldo',
alamat:'Jl.Mataram 3' })
$create (:Pembeli {id_member:3, nama:'Banu',
alamat:'Mantrijeron' })
$create (:Pembeli {id_member:4, nama:'Cantika',
alamat:'Ngestiharjo' })
```

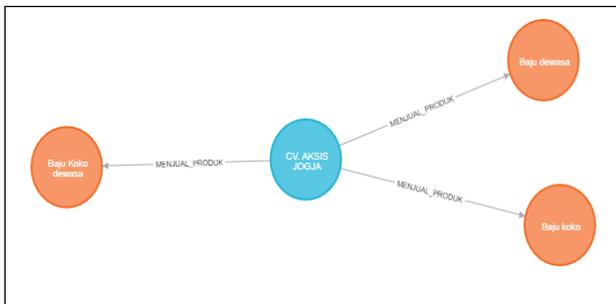
- Hasil proses query tersebut ditampilkan pada gambar 8



Gambar 8 Simpul Pembeli

C. Penghubung

Dalam rancangan basis data graph tidak semua tabel yang dihasilkan dalam basis data relasional dibuat suatu simpul. Hubungan antar simpul dapat dibuat suatu penghubung antar simpul. Gambar 9, simpul barang dengan simpul penjual (pelaku UMM) dan simpul barang yang dijual.



Gambar 9 Penghubung antar simpul

- Pembuatan Penghubung MEMBELI_PRODUK

```
$ match (u:Pembeli), (b:jual_produk)
where u.nama='Banu' and b.id_produk='p2'
create (u)-[:MEMBELI_PRODUK {tgl:['20/3/2020'],
banyak:[3] } ]->(b)
```

Pembuatan penghubung MEMBELI_PRODUK dilakukan untuk menyimpan data-data penjualan, seperti tanggal pembelian, banyak pembelian serta data lainnya

- Pembuatan Penghubung MENJUAL_PRODUK

```
$ match (p:UMKM), (b:jual_produk)
where p.id_umkm='u1' and b.id_produk='p2'
create (p)-[:MENJUAL_PRODUK
{produk:['baju']}]>(b)
$match (p:UMKM), (b:jual_produk)
where p.id_umkm='u1' and b.id_produk='p1'
```

```
create (p)-[:MENJUAL_PRODUK
{produk:['baju']}]>(b)
$ match (p:UMKM), (b:jual_produk)
where p.id_umkm='u1' and b.id_produk='p3'
create (p)-[:MENJUAL_PRODUK
{produk:['baju']}]>(b)
```

Pembuatan penghubung MENJUAL_PRODUK dilakukan untuk menyimpan data-data yang terkait dengan produk barang yang dijual oleh pelaku UMKM.

- Pembuatan Penghubung KATEGORI_PRODUK

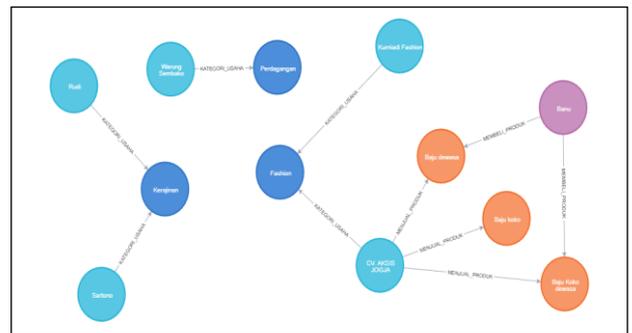
```
$match (u:UMKM), (b:kategori)
where u.id_umkm='u1' and b.id_kategori=1
create (u)-[:KATEGORI_USAHA
{produk:['Fashion'] } ]->(b)
$match (u:UMKM), (b:kategori)
where u.id_umkm='u3' and b.id_kategori=5
create (u)-[:KATEGORI_USAHA
{produk:['Sembako'] } ]->(b)
```

D. Uji coba hubungan Antar Simpul

Langkah pengujian dilakukan untuk mendapatkan berbagai informasi yang terkait dengan data-data yang disimpan dalam basis data graph. Hasil pengujian diantaranya

- Pengujian keterkaitan semua simpul yang sudah ada

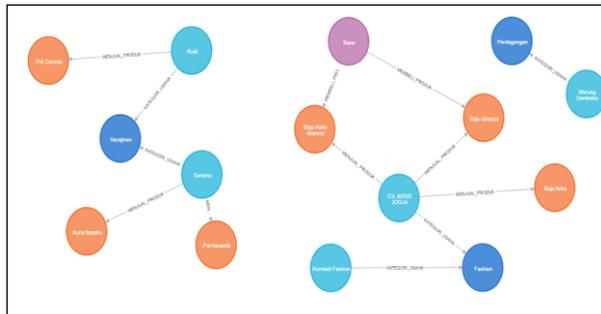
Pengujian ini untuk melihat keterkaitan semua simpul yang sudah ada. Gambar 11 memperlihatkan keterkaitan semua simpul yang ada. Dalam gambar 10, simpul UMKM sudah mempunyai simpul kategori tetapi belum ada transaksi penjualan, demikian juga simpul pembeli, belum semua simpul pembeli melakukan transaksi penjualan.



Gambar 10 Keterkaitan semua simpul yang ada

- Pengujian simpul yang saling terkait
 Pengujian ini melihat simpul-simpul yang saling terkait dalam penghubung. Gambar 11, menampilkan keterkaitan semua simpul dengan semua penghubung yang ada. Dari gambar tersebut belum semua simpul yang ada terdapat keterkaitan simpul, dalam gambar tersebut terdapat 2 simpul UMKM (CV Aksis Jogja dan Kurniadi Fashion), kedua simpul tersebut terhubung pada simpul kategori Fashion. Hubungan ini bertujuan untuk membuat kategori jenis usaha suatu UMKM.

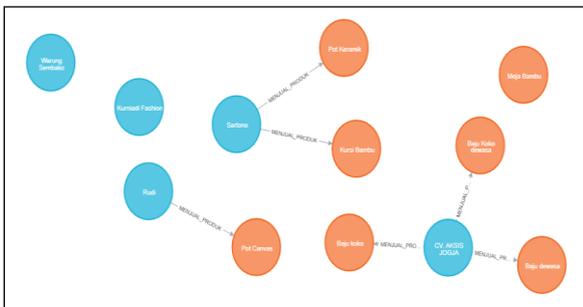
Hubungan menjual produk menyimpan data-data produk yang dijual UMKM. Dalam gambar simpul CV Aksis Jogja menjual produk baju dewasa, baju koko dewasa dan baju koko dewasa. Simpul lain yang dibuat hubungan adalah simpul Banu dengan simpul Baju dewasa dan baju koko dewasa. Informasi yang ada dalam hubungan simpul ini adalah pembeli Banu membeli 2 produk yang ada pada CV Aksis Jogja.



Gambar 11 Keterkaitan antar simpul

- Pengujian query untuk melihat pelaku UMKN yang sudah menjual produk.
\$ match (k:UMKM),(j:jual_produk) return k,j

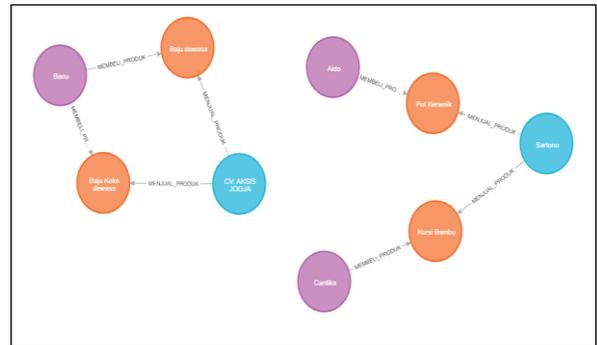
Query ini digunakan untuk melihat pelaku UMKM (simpul UMKM) yang ada berkaitan dengan simpul Jual_produk. Gambar 12, memperlihatkan hasil pengujian query. Query akan menghubungkan simpul UMKM dan simpul Jual_produk yang ada keterkaitan. Hasil query memperlihatkan masih ada beberapa simpul yang tidak saling terkait.



Gambar 11 Hasil query simpul UMKM dan simpul jual_produk

- Pengujian query untuk melihat pelaku UMKN yang sudah menjual produk.
\$match (p:Pembeli)-[m:MEMBELI_PRODUK]- (j:jual_produk)-[n:MEMBELI_PRODUK]- (u:UMKM) return p,m,j,u

Query ini digunakan untuk melihat data pembeli yang melakukan transaksi. Hasil query memperlihatkan simpul pembeli dikaitkan dengan simpul jual_produk dan simpul jual_produk dikaitkan dengan simpul UMKM. Gambar 12, memperlihatkan hasil pengujian query.



Gambar 12 Hasil query simpul Pembeli dan simpul UMKM yang dihubungkan dengan simpul Jual_produk

E. Uji coba hubungan Antar Simpul dalam bentuk tabel

Hasil query antar simpul di samping menghasilkan dalam bentuk graph juga dapat ditampilkan dalam bentuk teks atau tabel. Hasil pengujian yang dilakukan diantaranya

- Pengujian query untuk menampilkan data-data penjualan
\$ match (p:Pembeli)-[m:MEMBELI_PRODUK]- (j:jual_produk)-[n:MEMBELI_PRODUK]- (u:UMKM)
return p.nama,j.nama_produk, j.harga,m.tgl,u.nama_ukm

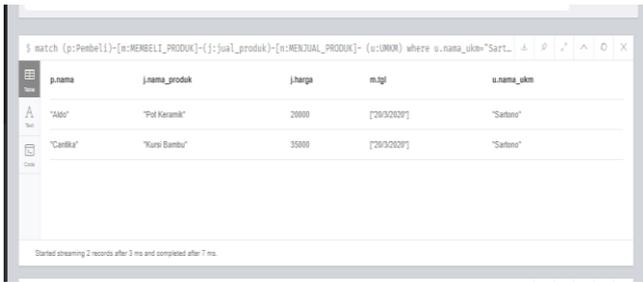
Proses query tersebut menghasilkan nilai luaran berupa data-data tertentu sesuai dengan nilai return. Gambar 13, menampilkan hasil query dalam bentuk teks. Dalam menampilkan teks nilai return dapat diatur sesuai dengan luaran yang diinginkan

p.nama	j.nama_produk	j.harga	m.tgl	u.nama_ukm
"Banu"	"Baju Koko dewasa"	30000	"[20200302]"	"CV AKSIS JOGJA"
"Banu"	"Baju dewasa"	50000	"[20200302]"	"CV AKSIS JOGJA"
"Sartono"	"Kura Bambu"	30000	"[20200302]"	"Sartono"
"Sartono"	"Pati Kerambi"	20000	"[20200302]"	"Sartono"

Gambar 13 Hasil Query dalam bentuk tabel

- Pengujian query untuk melihat pelaku UMKN yang sudah menjual produk dengan pelaku UMKM tertentu.
\$match (p:Pembeli)-[m:MEMBELI_PRODUK]- (j:jual_produk)-[n:MEMBELI_PRODUK]- (u:UMKM)
where u.nama_ukm="Sartono"
return p.nama,j.nama_produk,j.harga,m.tgl,u.nama_ukm

Query ini digunakan untuk melihat data pembeli yang melakukan transaksi dan dibatasi hanya untuk pembeli yang bernama Sartono. Hasil query memperlihatkan simpul pembeli dikaitkan dengan simpul jual_produk dan simpul jual_produk dikaitkan dengan simpul UMKM dan membatasi hanya pembeli yang bernama Sartono. Gambar 14, memperlihatkan hasil pengujian query.



Gambar 13 Hasil Query dalam bentuk tabel dengan pembatasan tampilan nama penjual

- Pengujian query untuk menampilkan semua data-data yang ada pada simpul penjual dan simpul UMKM yang ada keterkaitan simpul `match (p:Pembeli)-[m:MEMBELI_PRODUK]-(j:jual_produk)-[n:MENJUAL_PRODUK]-(u:UMKM) return p,m,j,u`

Hasil query menampilkan semua properti yang ada pada masing-masing simpul. Gambar 14, memperlihatkan hasil query dengan menampilkan semua properti yang ada pada simpul



Gambar 14 Hasil Query dalam bentuk tabel untuk semua properti yang ada pada setiap simpul

KESIMPULAN

Penggunaan basis data Graph dapat menjadi solusi untuk persoalan data yang besar. Dengan basis data graph, kelengkapan data yang disimpan tidak haru detail seperti dalam basis data relasional. Kelemahan basis data relasional yang setiap data harus disimpan dalam bentuk record dan disimpan dalam field akan menjadi persoalan jika data yang disimpan tidak lengkap. Dalam basis data relasional, isi field harus terisi. Dengan basis data Graph, kelengkapan data tidak menjadi keharusan.

Proses query untuk mendapatkan informasi dilakukan dengan proses query yang mirip dengan query yang ada pada SQL. Hasil basis data graph terdiri dari simpul UMKM, pembeli, produk, kategori dan simpul-simpul tersebut dilakukan keterkaitan dengan menggunakan penghubung diantaranya jual_produk, membeli_produk.

PENGHARGAAN

Penulis mengucapkan terima kasih kepada DRPM Direktorat Jenderal Riset dan Pengembangan yang telah memberi dukungan dana untuk kegiatan penelitian pada skema penelitian terapan unggulan perguruan tinggi tahun pendanaan 2019 dan 2020 serta Dinas Komunikasi dan Informatika kabupaten Bantul yang telah memberikan

kesempatan kepada tim IST AKPRIND untuk memberikan pendampingan.

REFERENSI

- [1] P. Wilton and J. W. Colby, *Beginning SQL*. Indianapolis: Wiley Publishing, Inc, 2005.
- [2] S. Medhi and H. Baruah, 'Relational basis data and graph basis data: A comparative analysis', *Journal of Process Management*. New Technologies, vol. 5, no. 2, pp. 1–9, 2017, doi: 10.5937/jouproman5-13553.
- [3] C. Strauch, *NOSQL basis datas*. 2010.
- [4] F. A. Bhaswara, R. Sarno, and D. Sunaryono, 'Perbandingan Kemampuan Basis data NoSQL dan SQL dalam Kasus ERP Retail', *Jurnal Teknik ITS*, vol. 6, no. 2, pp. 510–514, 2017, doi: 10.12962/j23373539.v6i2.24031.
- [5] J. Dai, 'SQL to NoSQL: What to do and How', *IOP Conference Series: Earth and Environmental Science*, vol. 234, no. 1, 2019, doi: 10.1088/1755-1315/234/1/012080.
- [6] R. Čerešňák and M. Kvet, 'Comparison of query performance in relational a non-relation basis datas', in *13th International Scientific Conference on Sustainable, Modern and Safe Transport Slovak Republic*, May 29-31, 2019 (TRANSCOM 2019), 2019, pp. 170–177, doi: 10.1016/j.trpro.2019.07.027.
- [7] A. Sharma, 'Difference between SQL and NoSQL', *International Journal of Management, IT & Engineering*, vol. 8, no. 6, pp. 444–452, 2018.
- [8] B. M. Sasaki, J. Chao, and R. Howard, 'Graph Basis datas For Beginners'.
- [9] A. Kadir, *Dasar Basis Data Graft & NEo4j*. Yogyakarta: Penerbit Andi, 2020.
- [10] S. Patil, G. Vaswani, and A. Bhatia, 'Graph Basis datas- An Overview', *International Journal of Computer Science and Information Technologies*, vol. 5, no. 1, pp. 657–660, 2014.
- [11] M. F. P. F, K. R. Reshma, and S. M. Varghese, 'O UTCOME A NALYSIS U SING N EO 4 J G RAPH', *International Journal on Cybernetics & Informatics (IJCI)*, vol. 5, no. 2, pp. 229–236, 2016, doi: 10.5121/ijci.2016.5225.
- [12] W. Khan, E. ahmed, and W. Shahzad, 'Predictive Performance Comparison Analysis of Relational & NoSQL Graph Basis datas', *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 5, 2017, doi: 10.14569/ijacsa.2017.080564.
- [13] H. R. Vyawahare, P. P. Karde, and V. M. Thakare, 'An efficient graph basis data model', *International Journal of Innovative Technology and Exploring Engineering*, vol. 88, no. 10, pp. 1292–1295, 2019, doi: 10.35940/ijitee.I7805.0881019.
- [14] C. M. Widyayanti and A. I. Nurhidayat, 'No Title', *Jurnal Manajemen Informatika*, vol. 6, no. 2, pp. 14–22, 2016.
- [15] E. Zhang, J. Fiaidhi, S. Mohammed, O. Rd, T. Bay, and O. N. Pb, 'Social Recommendation using Graph Basis data Neo4j : Mini Blog , Twitter Social Network Graph Case Study', *International Journal of Future Generation Communication and Networking*, vol. 10, no. 2, pp. 9–20, 2017.
- [16] S. A. T. Mpinda, L. G. Maschietto, M. T. S. Prado, and M. X. Ribeiro, 'Graph basis data application using Neo4j: Railroad planner simulation', *ICEIS 2015 - 17th International Conference on Enterprise Information Systems, Proceedings*, vol. 1, no. 04, pp. 399–403, 2015, doi: 10.5220/0005469003990403.
- [17] P. W. Wirawan, D. E. Riyanto, D. Mutiara, and K. Nugraheni, 'Graph Basis data Schema for Multimodal Transportation in Semarang', *Journal of Information Systems Engineering and Business Intelligence*, vol. 5, no. 2, pp. 163–170, 2019.
- [18] Wahyudi and Fajril Akbar, 'Ekstraksi Basis Pengetahuan Ke Dalam Basisdata Graf Menggunakan Graf Properti', *Jurnal Nasional Teknologi dan Sistem Informasi*, vol. 5, no. 1, pp. 41–48, 2019.
- [19] P. W. Wirawan, D. E. Riyanto, and K. Khadijah, 'Pemodelan Graph Basis data Untuk Moda Transportasi Bus Rapid Transit', *Jurnal Informatika*, vol. 10, no. 2, pp. 1271–1279, 2016, doi: 10.26555/jifo.v10i2.a5072.
- [20] R. Y. Rachmawati and M. Sholeh, 'Design and Analysis of Basis data Clustering System of Micro , Small and Medium Enterprises (MSME) in Bantul Regency Design and Analysis of Basis data Clustering System of Micro , Small and Medium Enterprises (MSME) in Bantul

- Regency', Journal of Physics: Conference Series PAPER, vol. 1413, no. 1, 2019, doi: 10.1088/1742-6596/1413/1/012028.
- [21] D. Hardiyanto, M. Sholeh, and R. Y. Rachmawati, 'Optimization of information technology utilization based on geography information system as a marketing media for micro , small , and medium enterprises to improve competitiveness in Bantul Regency , Indonesia Optimization of information technology utiliz', IOP Conference Series: Materials Science and Engineering PAPER, vol. 807, 2019, doi: 10.1088/1757-899X/807/1/012010.